

NASA Contractor Report 195451

1N-39
~~44580~~
61P

ORIGINAL CONTAINS
PAPER ILLUSTRATIONS

On the Finite Element Implementation of the Generalized Method of Cells Micromechanics Constitutive Model

T.E. Wilt
University of Akron
Akron, Ohio

March 1995

Prepared for
Lewis Research Center
Under Cooperative Agreement NCC3-368



National Aeronautics and
Space Administration

(NASA-CR-195451) ON THE FINITE
ELEMENT IMPLEMENTATION OF THE
GENERALIZED METHOD OF CELLS
MICROMECHANICS CONSTITUTIVE MODEL
Final Report (Akron Univ.) 61 p
N95-24398
Unclass
G3/39 0044580

On The Finite Element Implementation of The Generalized Method of Cells Micromechanics Constitutive Model

T. E. Wilt

Department of Civil Engineering
The University Of Akron
Akron Ohio 44325-0604

Abstract

The Generalized Method of Cells (GMC), a micromechanics based constitutive model, is implemented into the finite element code MARC using the user subroutine HYPELA. Comparisons in terms of transverse deformation response, micro stress and strain distributions, and required CPU time are presented for GMC and finite element models of a fiber/matrix unit cell. GMC is shown to provide comparable predictions of the composite behavior and requires significantly less CPU time as compared to a finite element analysis of the unit cell. Details as to the organization of the HYPELA code are provided with the actual HYPELA code included in the appendix.

1.0 Introduction

In the design of aerospace components, the use of composite materials such as metal matrix composites (MMC) are being investigated as a means of obtaining increased operating loads, improved durability, and decreased weight. For example, MMC's are being considered for critical engine components, such as rotors, blades and nozzles which are subjected to high temperature environments under complex load histories. From the analysis standpoint, the use of nonlinear finite element methods will be required for the

design and structural response predictions of these components. In addition, there is increased interest in developing life prediction methods to predict the service life of these components which will also require accurate finite element analysis.

When analyzing a structure using the finite element method a key ingredient is the constitutive model as it determines how accurate the global response of the structure will be predicted. For structures composed of isotropic materials, there is considerable experience with plastic and viscoplastic constitutive models and their implementation into the nonlinear finite element solution. On the other hand, the development of effective constitutive models for the nonlinear material response of a composite is relatively new. The primary goal in formulating a constitutive model for composite materials is the development of the relationship between the macro strain and the macro stress, i.e. the effective macro stiffness. There are two approaches [2] in modeling the composite material: 1) a macro, equivalent, homogeneous material model, 2) a micromechanics based model. Considering the operating environment that the components will be subjected to, using a micromechanics based material model may be necessary in order to include the many nonlinear mechanisms that occur within the composite, such as, fiber/matrix debonding, localized matrix yielding, matrix cracking, oxidation effects, etc.. These type of effects are believed to be especially important for the life analysis of the component. There has been debate whether or not using a micromechanics based constitutive model in the finite element method is feasible with regards to computational expense. This point will be addressed in the following section by examining some of the various approaches currently available.

2.0 Computational Aspects

The general solution algorithm for a nonlinear finite element analysis will be discussed in terms of the computational effort required at both the global (structural) level and the local (element) level. This discussion will be used to show how the use of micromechanics based models may or may not significantly increase the computational expense of a nonlinear finite element analysis.

The nonlinear finite element equilibrium equation takes the form,

$$[K]^T \{\Delta U\} = \{\Delta R\} \quad (1)$$

where $[K]^T$ is the structural tangent stiffness matrix and the term $\{\Delta R\}$ is the out-of-balance force vector, i.e.,

$$\{\Delta R\} = \{F\} - \sum_{l=1}^n \int_{-1}^1 \int_{-1}^1 \int_{-1}^1 [B]_l^T \{\sigma\}_l |J| \frac{h_l}{t_c} dt_l dr ds \quad (2)$$

Global equilibrium is satisfied by iterating on eqn. (1) until the out-of-balance force vector $\{\Delta R\}$ is effectively zero (within some predefined convergence tolerance). This global equilibrium equation therefore constitutes the first level of iterations.

From the out-of-balance vector $\{\Delta R\}$, eqn. (2), the integral is evaluated at each integration point in the structure. Specifically, at each integration point, the global finite element solution provides the current strain state and the current element stress σ is obtained from the evaluation of the local constitutive model. That is, the constitutive model calculates the current stress and should provide the current material stiffness at that point. Thus a micromechanics model needs to be able to take the macro/composite strain and, after performing a local stress/strain analysis on the RVE, provide the current macro/composite stress state and equivalent composite material stiffness. Having the current equivalent composite material stiffness will allow the use of a tangent stiffness matrix in the global equilibrium equation (1). The advantage of using a tangent stiffness is the increased rate of convergence in the global equilibrium iterations, with a quadratic rate obtainable if a properly formulated local tangent stiffness is utilized.

It is apparent that the next important level of computations is at the integration point. Here, one may or may not encounter local level iterations which will depend on the type of constitutive model being used. For clarity in the following discussion, assume that the material at the integration point in question is isotropic.

If for example, a classical plasticity model is used, the radial return algorithm, which is commonly used, requires local level iterations in order to return the stress to the yield surface, an example of which is given in [11]. Another class of material models would be the continuum-based viscoplastic models. These rate dependent models contain flow and evolution equations

which must be integrated in time. This integration is performed using either an explicit or implicit integration method. If an explicit integrator is used, for example forward Euler, no additional local iterations are required. But it is well documented that such explicit methods are conditionally stable and may require small global time steps in order to achieve convergence. Solving a highly nonlinear structural problem with many small time steps becomes computationally expensive. As a result, the use of implicit integrators are increasingly being used since they are far more robust for integrating the numerically "stiff" viscoplastic rate equations and provide unconditional stability, thus allowing larger time steps to be used. But, due to their implicit formulation, local iterations are required.

From the above discussion, it is apparent that two levels of iterations are typically required to solve highly nonlinear finite element problems: 1) global equilibrium, and 2) local constitutive model iterations. These multiple levels of iterations are what makes nonlinear finite element solutions inherently computationally expensive. Therefore, it is critical that any micromechanics constitutive model be accurate yet efficient.

Any micromechanics method which is based upon a finite element RVE becomes in effect a finite element analysis within a finite element analysis. A complete nonlinear finite element analysis must be performed to take the current macro strain state provided by the finite element solution and produce the current stress state in the RVE. Note that at this point, only the micro stress/strain states exist. Additional computations are still required to convert the micro (constituent) stresses to an equivalent macro composite stress state and calculate the equivalent macro material stiffness. An example of a finite element based method is that presented by Wu et. al. [15]. In this method a RVE for a periodic hexagonal array (PHA) finite element model is used to perform the micromechanics analysis and calculate the instantaneous macro composite material stiffness. The PHA model in effect serves as the local constitutive model and is called at each integration point in the structure. The ultimate objective of a micromechanics analysis is the calculation of the equivalent macro material stiffness. In this regard, a so-called equivalent homogeneous volume, EHV, is introduced. The EHV is defined by the equivalent macro instantaneous material parameters which are considered as unknowns. These material parameters are determined by applying the same periodic boundary conditions to the EHV as are applied to the RVE. Iterations are then performed until equilibrium is attained such

that the EHV and RVE have the same total energy change. One can see that the level of computational effort is increased. In addition, each finite element in this RVE has a constitutive model which must be evaluated and, as discussed above, may or may not require local iterations in its calculations. It is stated in [15] that accurate calculation at each material point requires substantial effort which would need to be repeated many times in a typical nonlinear finite element analysis.

Another, more general, finite element based approach is that of Fish and Wagiman [7] which is applicable to both periodic and non-periodic domains. That is, the solution or the microstructure may be periodic or non-periodic. In this approach, the homogenized material properties are based upon the displacement field of the unit cell which is obtained by solving the finite element model of the unit cell for the six different loading conditions. In addition, in this "multiscale" finite element method, two finite element meshes may be used. A "macro" mesh is used for the global problem which is the part of the structure where the composite microstructure and the solution is periodic. A second "micro" mesh is used for the local problem where either the solution or the microstructure is not periodic. This produces a coupled global/local problem which, as described in [7], requires an iterative solution process.

From the above discussion, it is apparent that while the use of finite element based micromechanics models provide significant capabilities, the associated computational expense (time and memory) and complicated algorithms make them less attractive. The use of parallel computers and parallelized/optimized code [13] in order to decrease computation time limits the usefulness of the method since such sophisticated computer resources are not commonly available.

In order to reduce the computational expense, an efficient micromechanics model should be more analytical in nature in which closed form expressions are available for the macro/micro relationships. This class of models are based upon micromechanics but include various assumptions in order to derive semi-analytical expressions for the effective composite stiffness and relationships between macro and micro stress and strain.

Svobodnik et. al. [12] use the vanishing fiber diameter model of Dvorak and Bahei-El-Din [6] in the context of the finite element method. In the vanishing fiber model, the fibers are assumed to behave elastically to failure and any effects of fiber interaction are neglected. The result is a simple

model which gives the equivalent macro response of the composite expressed in terms of the constituent properties and their volume fractions. Svobodnik [12] demonstrates that the model may be implemented into the ABAQUS finite element code with relative ease. There are limitations of the vanishing fiber model. The first is neglecting any fiber interaction effects. It has been shown that packing arrangement, i.e. square, hexagonal, etc., does have an effect on the transverse response of the composite [3]. A second limitation of the model is the inability to capture specific micro stress/strain distribution effects such as localized yielding in the matrix [12] due to its assumption of homogeneous stress and strain fields in the fiber and matrix. Such a limitation is important if accurate micro stress and strain is necessary as in the case of damage/life analysis of the composite. With these limitations, it is believed that any benefit of using a micromechanics based model have been lost. The choice of micromechanics-based models is either a simplified model which provides the necessary macro composite stress strain relation but because of its simplified form is not able to provide micro stress strain information in sufficient detail. On the other hand, finite element based micromechanics models do provide detailed (depending on mesh density) fiber and matrix stress and strain distributions but at the cost of significant computational expense in terms of the microlevel analysis and the calculation of the current macro composite material stiffness. Thus the "best" micromechanics model would be a combination of the above two approaches, that is, it would provide sufficiently detailed microlevel stress and strain distributions and a macro composite material stiffness in an efficient manner.

In this regard, the Generalized Method of Cells (GMC) micromechanics model of Paley and Aboudi [9] is believed to satisfy the above requirements. GMC is an extension of Aboudi's earlier method of cells [1] in that GMC now allows for more than 4 subcells to describe the fiber and matrix phases. By having a variable number of subcells provides significant flexibility in modeling a variety of RVE's, for example different fiber shapes, fiber packing arrangements, inclusion of a fiber/matrix interface layer, etc. Some typical RVE's constructed using GMC are shown in Figure 1. In addition, one of the noteworthy features of the GMC model is that it provides an analytical constitutive expression for the composite as well as macro-micro relations for stress and strain. Thus no iterations are required, as in a finite element based model, to obtain the current macro composite material stiffness. Yet, as will be shown in the following sections, GMC provides detailed microlevel

stress and strain distributions which are comparable to those obtained from a finite element RVE analysis.

Some details of GMC pertaining to its implementation in the HYPELA subroutine in MARC will be briefly outlined below. Additional theoretical details may be found in [9].

3.0 Generalized Method of Cells: The HYPELA Subroutine

As mentioned in section 1, one of the uses for an efficient micromechanics model would be in the finite element method. In order to access the requirements of a micromechanics model in terms of CPU time and overhead, i.e. memory/storage requirements, the GMC model has been implemented into the finite element code MARC through the use of the user subroutine HYPELA [8]. The subroutine HYPELA is provided for users to implement their own constitutive models into MARC. The implementation of GMC into a HYPELA format was relatively straightforward. The main subroutine is HYPELA which contains all of the micro and macro relationships that comprise GMC. A branch is made to the desired constitutive model for each subcell material. The equations below are presented in the same order in which they appear in the HYPELA subroutine. One of the key parts of GMC is the formation of the A-matrix. This matrix consists of submatrices which contain quantities defining the material constants and the geometry of the individual subcells. The A-matrix is partitioned in the HYPELA subroutine as follows,

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_M & \mathbf{0} & \mathbf{A}_M \\ \mathbf{A}_G & \mathbf{J} & \mathbf{0} \end{bmatrix} \quad (3)$$

Let the submatrix $\mathbf{A}^*$ be defined as,

$$\mathbf{A}^* = \begin{bmatrix} \mathbf{A}_M \\ \mathbf{A}_G \end{bmatrix} \quad (4)$$

where $\mathbf{A}^*$ is a square matrix with dimensions $N_\beta \times N_\gamma$. Also define $\mathbf{K}$ as,

$$\mathbf{K} = \begin{bmatrix} \mathbf{0} \\ \mathbf{J} \end{bmatrix} \quad (5)$$

and A^P as

$$A^P = \begin{bmatrix} A_m \\ 0 \end{bmatrix} \quad (6)$$

In the above matrices, the matrix A_M involves the elastic material properties of the subcell and the matrix A_G involves the geometry of the subcell and the matrix J relates the average composite strain to the average subcell strain.

The only necessary modification to the code included in Appendix A is the linking of a matrix routine for the inversion of the A matrix. It will be noted that in the present code, two linpack subroutines, i.e. SGEFA and SGESL, were originally used for this purpose. The calls to these subroutines are left in the code to show the proper location for the matrix inversion subroutine call. Also note that since only the elastic material constants are being used, it is only necessary to form the A matrix at the initial time step. On the other hand, if viscoplasticity (tangent stiffness) or damage effects are to be included in the stiffness, the A matrix would need to be calculated at each time step.

Once the A -matrix has been formed, the effective macro composite elastic stiffness may be calculated from,

$$\bar{C} = \frac{1}{hl} \sum_{\beta=1}^{N_\beta} \sum_{\gamma=1}^{N_\gamma} h_\beta l_\gamma C^{(\beta\gamma)} A^{(\beta\gamma)} \quad (7)$$

in which $C^{(\beta\gamma)}$ is the subcell elastic stiffness. Note that calculating the critical macro stress/strain relationship necessary for the global finite element solution only required straightforward matrix multiplication operations.

Another advantage of GMC is that the inelastic response of the constituents may be modeled using either a plastic or viscoplastic constitutive model. The quantity $\dot{\epsilon}^{I(\beta\gamma)}$ is the inelastic strain rate in the subcell and calculated by calling the appropriate constituent constitutive model subroutine. In the current implementation, an isothermal Bodner-Partom [5] viscoplastic material model is included. Additional material models can be included with relative ease. Note that both the viscoplastic material constants for the Bodner-Partom model and the elastic material constants are defined in the HYPELA/GMC subroutine. The viscoplastic constants are stored in the two dimensional array VPROB and currently is dimensioned for two different materials having a maximum of 12 viscoplastic constants per material. Note that the constitutive model calculates the inelastic strain rate for each

subcell. That is the matrix subcell inelastic strain, $\dot{\epsilon}^{I(\beta\gamma)}$, is calculated from the flow law,

$$\dot{\epsilon}^{I(\beta\gamma)} = D_0 \exp \left\{ -\frac{1}{2} \left(\frac{(Z^I + Z^D)^2}{3J_2} \right)^n \right\} \frac{s_{ij}}{\sqrt{J_2}} \quad (8)$$

From the individual subcell inelastic strain rates, the macro composite inelastic strain rate, $\dot{\epsilon}^I$, is calculated from,

$$\dot{\epsilon}^I = C^{-1} \frac{1}{hl} \sum_{\beta=1}^{N_\beta} \sum_{\gamma=1}^{N_\gamma} h_\beta l_\gamma C^{(\beta\gamma)} [\mathbf{A}^{P(\beta\gamma)} \dot{\epsilon}_s^P - \dot{\epsilon}^{P(\beta\gamma)}] \quad (9)$$

and is stored in the array EPG.

The average subcell total strain rate, $\dot{\epsilon}^{(\beta\gamma)}$, is stored in the array ES and is calculated from the relation,

$$\dot{\epsilon}^{(\beta\gamma)} = \mathbf{A}^{*-1(\beta\gamma)} \mathbf{K} \dot{\epsilon} + \mathbf{A}^{P(\beta\gamma)} \dot{\epsilon}_s^I \quad (10)$$

The average subcell stress rate, $\dot{\sigma}^{(\beta\gamma)}$, is stored in the array SS and is found from

$$\dot{\sigma}^{(\beta\gamma)} = C_{ijkl}^{(\beta\gamma)} (\dot{\epsilon}_{kl}^{(\beta\gamma)} - \dot{\epsilon}_{kl}^{I(\beta\gamma)}) \quad (11)$$

where $C_{ijkl}^{(\beta\gamma)}$ is the elastic material stiffness of the $(\beta\gamma)$ subcell.

The macro composite strain rate, $\dot{\epsilon}$ is stored in the array EG and is calculated from the solution of the global (structural) finite element solution. Note that in HYPELA, MARC provides the current strain increment in the array DE and the rate is simply obtained by dividing each component by Δt . Finally, the macro composite stress, $\dot{\sigma}$, is obtained from the expression,

$$\dot{\sigma} = \bar{C}(\dot{\epsilon} - \dot{\epsilon}^I) \quad (12)$$

and is stored in the array SG.

Before leaving the subroutine HYPELA it is necessary to integrate the above rate equations. Presently, the calculations in HYPELA are performed in a sequence that is consistent with a simple forward Euler integration scheme. The current solution at time $t + \Delta t$ is found from,

$$x^{t+\Delta t} = x^t + \Delta t \dot{x}|_t \quad (13)$$

where x is an array containing all of the macro and micro variables and the array $\dot{x}$ contains the rates which are evaluated based upon the previous converged solution at time t . In the subroutine HYPELA the array SIGAL is used to store the current total values of the stress, strain and internal state field variables, while the array DSIGAL stores the rate of the stress, strain and internal state field variables. These arrays are arranged identically. The first 37 positions are used to store the macro (composite) state variables which consist of total strain (6), Cauchy stress (6), inelastic strain (6), room for two internal state variables (12), thermal strains (6) and current temperature, in this order. The micro (subcell) quantities are then stored in "blocks" of 36 which are organized according to the description given above for the macro quantities with the exception that no temperature variable is stored. Presently, the arrays are dimensioned for a maximum of 49 subcells.

When the user subroutine HYPELA is used, MARC solves for global equilibrium using the initial strain form of the equilibrium equation, i.e.,

$$[K^e] \{\Delta U\}^{(i)} = \{F\} - \sum_{l=1}^n \int_{-1}^1 \int_{-1}^1 \int_{-1}^1 [B]_l^T \{\sigma\}_l |J| \frac{h_l}{t_c} dt_l dr ds - \int_V [B]^T [C]^e \{\Delta \epsilon\}^I dV \quad (14)$$

where $[K^e]$ is the global elastic stiffness. As a result, the macro composite stress is placed in the vector S, the current inelastic strain increment is placed in the vector G, and the macro composite elastic stiffness is placed in the array D so that they may be passed back to the main MARC program.

4.0 Generalized Method of Cells Evaluation

In order to access the accuracy of GMC, a series of comparisons were performed with finite element meshes representing the fiber and matrix unit cell. The finite element meshes used are shown in Figure 2. Material constants for a SiC fiber and Ti-15-3 matrix at 426C were used in the present analysis. The unit cell was subjected to a tensile load under strain control with the strain rate being 0.001/s to a maximum macro transverse strain of 0.02.

4.1 Macrolevel: Deformation Response

The results for longitudinal response were identical for all of the finite element meshes and the GMC models. This is to be expected since even a simple rule of mixtures model can adequately predict the longitudinal response. However, the transverse response requires more accurate modeling of the unit cell. The transverse deformation response for the composite is shown in Figure 3.

From Fig. 3, the 4 finite element mesh grossly over-predicts by approximately 42% the transverse response as compared to the 4-cell GMC model. The response for the 16 finite element mesh is approximately 4% too stiff, while the response for the 64 finite element mesh is approximately 2% softer. Note however that the 49-cell GMC model gives a softer response than the 4-cell model by approximately 8%. As a result, a 196-cell GMC model was analyzed and its response matched that of the 49-cell model. Thus the 49-cell model has converged to the correct transverse response of the composite. The 272 finite element mesh is approximately 5% stiffer than the 49-cell GMC model. Finally, a finite element mesh of 1088 elements was used to match the response of the 49-cell GMC model. A note should be made that it may be possible to match the 49-cell response using a smaller finite element mesh. As may be noted from figure 2, the 1088 mesh was obtained by simply refining the 272 mesh by subdividing each element into four elements. Thus the 1088 element mesh may not be the "optimal" mesh using the minimum number of elements required to converge with the 49-cell GMC model. In any case, a finite element mesh larger than 272 elements is required to match the response of a 49-cell GMC model. The implications of requiring such a large finite element model will be discussed in section 4.3.

4.2 Microlevel: Stress and Strain

The other aspect of choosing a micromechanics model is how accurate the microlevel fields are predicted. Of particular interest are the stress and inelastic strain distributions. Figure 4 shows the effective stress, J_2 , distribution for the 49-cell model and the 64 finite element mesh. Note how the distributions compare in both distribution and magnitude. Figure 5 shows the effective inelastic strain distribution for the GMC and finite element models. Again, the distribution and magnitudes are comparable. One interesting note is with regards to the in-plane shear stress distribution. Inherent in the GMC method is the uncoupling of the shear and normal stress components. That is, for a given applied normal stress, a shear stress will not be produced.

As a result, there is a state of zero shear stress in the unit cell for the present tensile test. On the other hand, there is a shear stress present in the finite element solution. Figure 6 shows the shear stress distribution for the finite element model. It is interesting to note that even with the absence of any shear stress component, GMC still accurately predicts the macro transverse deformation response. This maybe attributed to the fact that the effective stress distribution does match that of the finite element solution [4]. Since the viscoplastic model is "driven" by J_2 , the deformation response should be comparable. However, the implications of GMC's lack of shear stress distribution may surface when considering other aspects such as damage modeling. For example, fiber/matrix debonding which may be a function of the local shear stress in the interface region. Such issues are currently being addressed in ongoing research.

4.3 Computation Time Comparisons

The goal is to be able to analyze composite structures at both the macro and micro levels [2]. That is, to analyze the response of the structure (macrolevel) and then, at various points in the structure, go to the microlevel and determine the stress, strain and other internal state variables in the constituents of the composite. This type of macro/micro analysis capability is believed to be important when trying to capture the damage mechanisms occurring at the fiber/matrix level which in turn affect the life of the structure.

In the context of the finite element method, these micro analyses would be performed at each integration point. Recall that when forming the finite element local stiffness matrix, a constitutive model is called at each integration point where for a given strain state, the corresponding updated stress state and the material stiffness is returned. Thus, the 49-cell GMC model, contained in HYPELA, would be evaluated at each integration point. Similarly, one of the finite element models, either the 272 or 1088 element mesh depending on the desired accuracy, would have to be evaluated at each integration point.

To study GMC, only a single finite element was used. Therefore, the total CPU time was then divided by eight (eight integration points per element) to obtain the approximate CPU time at one integration point. On the other hand, since the finite element model of the fiber/matrix unit cell would have to be analyzed at each integration point, the total CPU time was used for comparison to GMC.

With the above in mind, direct comparisons in terms of CPU time with the 4, 16, 64, 272 and 1088 finite element models of the fiber/matrix unit cell and the 4 and 49 cell GMC models were performed. A second reason for using a 49-cell model was that if other RVE's are used to represent other fiber architectures (hexagonal pack, interface layer, etc.) the GMC RVE would consist of 49 cells on average [3].

All of the results presented in the Tables were generated on the Cray-YMP at NASA Lewis Research Center. Also, all of the results presented have been normalized with respect to the CPU time for the GMC model under comparison. In Table 1, if the 49-cell GMC model requires 1 CPU unit, a finite element analysis using 1088 elements that provides the equivalent accuracy, in terms of transverse displacement, requires approximately 3550 CPU units. This clearly demonstrates the penalty one has to pay in terms of the overhead required for the finite element method and the substantial savings in CPU time the GMC method provides. One must also realize that the solution times reported here are only for the stress analysis portion of the unit cell. Determining the current equivalent inelastic macro and micro properties has not been performed. This phase will require what is believed to be significant computation time. Further details of the calculations that would be required in this phase may be found in [14].

An alternative approach to the above micromechanics models are the phenomenological continuum-based models. This class of models treat the composite as an homogenous anisotropic material in which no distinction is made between the fiber and matrix constituents. Therefore no micro stress-strain information is available. One of the arguments for using such models is that they provide the necessary macro relationships required in a structural analysis and are believed to require the minimum amount of computation time. For comparison purposes, Robinson's viscoplastic, transversely isotropic continuum-based model [10] was characterized for the SiC/Ti-15-3 composite. The viscoplastic model was then implemented into MARC through the HYPELA subroutine. Again as in the case of the GMC model, a single finite element was used and the total CPU time was divided by eight to obtain the approximate CPU time for one integration point. The same tensile strain control test, as described above, was performed. Comparisons of the CPU time for the GMC, finite element and continuum-based models are given in Table 2. Note that the continuum model is 20% faster than the 4-cell GMC model and 80% faster than the 49-cell GMC model. Of course,

one needs to keep in mind that the GMC models provide additional information on the micro (constituent) level that the continuum-based models do not provide.

5.0 Conclusions

The Generalized Method of Cells micromechanics model has been implemented into the finite element code MARC through the use of the user subroutine HYPELA. The implementation was found to be rather straightforward requiring no more effort than other implementations of "standard" constitutive models, for example the continuum-based viscoplastic model of Robinson which was also used in a HYPELA subroutine for this work.

It was found that the finite element method requires a significant number of elements in order to adequately predict the transverse deformation behavior of the composite. As a result, the computation time far exceeds that required by the GMC model. An important point is that the CPU time reported for the GMC model includes the calculations for the necessary macro stress-strain relations, i.e. current RVE material stiffness. While CPU times reported for the present finite element analyses are only for performing the nonlinear stress analysis on the RVE. Additional calculations, producing larger CPU times, are required to obtain the current RVE material stiffness. The speed of GMC is attributed to the analytical expressions for the macro/micro stress strain relationships and for the macro composite stiffness.

The GMC model has been shown to provide accurate effective stress, (J_2 form), stress and inelastic strain distributions. One important observation is the lack of any micro shear stress components for the GMC model. As noted, this does not appear to affect the deformation response predictions of GMC, but how this limitation would affect particular life prediction damage models, which may depend on specific micro stress components, needs to be investigated further. For example, how would an interfacial debonding criteria, which may be a function of the local in-plane shear stress, be affected? Research in the area of life analysis is currently in progress.

6.0 Acknowledgement

This work was performed under NASA cooperative grant NCC3-368. The grant monitor is Dr. Steven M. Arnold.

7.0 References

1. Aboudi, J., "Mechanics of Composite Materials: A Unified Micromechanical Approach", Elsevier, 1991.
2. Arnold, S.M. and Castelli, M.G.: "Continuum Based Theoretical and Experimental Studies in Deformation and Damage of MMCs at NASA-Lewis: Progress and Trends", *Comp. Engng.*, Vol. 4, No. 8, pp. 811-828, 1994.
3. Arnold, S.M., Wilt, T.E., and Pindera, M.J.: "Impact of Fiber Architecture on the Inelastic Response of Metal Matrix Composites Via the Generalized Method of Cells (GMC)", *HITEMP Review 1994*, NASA CP 10146, pp. 30:1-14, 1994.
4. Arnold, S.M. Wilt, T.E., Saleeb, A.F., and Castelli, M.G.: "An Investigation of Marco and Micromechanical Approaches for a Model MMC System", *HITEMP Review 1993*, NASA CP 19117, pp. 52:1-12, 1993.
5. Bodner, S.R. and Partom, Y.: "Constitutive Equation for Elastic-Viscoplastic Strain Hardening Materials", *Jnl. Appl. Mech.*, Vol. 42, pp. 385- , 1975.
6. Dvorak, G. J., and Bahei-El-Din, Y. A., "Plasticity Analysis of Fibrous Composites", *J. Appl. Mech.*, Vol. 49, pp.327-335, 1982.
7. Fish, J. and Wagiman, A. "Multiscale Finite Element Method for a Locally Nonperiodic Heterogeneous Medium", *Comp. Mech.*, 12, pp. 164-180, 1993.
8. MARC K6 User Subroutines/Special Routines, Volume D, MARC Analysis Research Corporation, Palo Alto, California, 1994.
9. Paley, M. and Aboudi, J., "Micromechanical Analysis of Composites by the Generalized Cells Model," *Mechanics of Materials*, Vol. 14, pp. 127-139, 1992
10. Robinson, D. N., and Duffy, S. F., "Continuum Deformation Theory for High Temperature Metallic Composites", *J. Eng. Mech.*, vol. 116, n. 4, pp. 832-844, 1990.

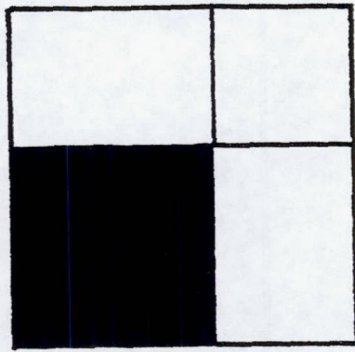
11. Simo, J. C. and Taylor, R. L., "A Return Mapping Algorithm for Plane Stress Elastoplasticity", *Int. J. Num. Meth. Engng.*, Vol. 22, pp. 649-670, 1986.
12. Svobodnik, A. J., Bohm, H. J., and Rammerstorfer, F. G., "A 3/D Finite Element Approach For Metal Matrix Composites Based On Micromechanical Models", *Int. J of Plasticity*, Vol. 7, pp. 781-822, 1991.
13. Swan, C. C., "Techniques for Stress- and Strain-controlled Homogenization of Inelastic Periodic Composites", *Comp. Meth. Appl. Mech. Engng.*, 117, pp 249-267, 1994.
14. Teply, J. L. and Reddy, J. N., "A Unified Formulation of Micromechanics Models of Fiber-Reinforced Composites", in *Inelastic Deformation of Composite Materials*, IUTAM Symposium, Troy, New York, May 29 - June 1, 1990.
15. Wu, J. F., Shepard, M. S., Dvorak, G. J., and Bahei-El-Din, Y. A., "A Material Model for the Finite Element Analysis of Metal Matrix Composites", *Composites Science and Technology*, Vol. 35, pp. 347-366, 1989.

Table 1: CPU Time Comparisons

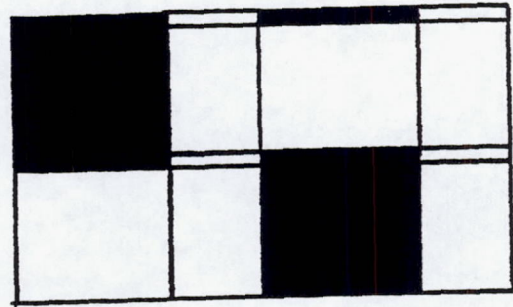
GMC		Finite Element					
case	CPU time	mesh size →	4	16	64	272	1088
4-cell	1.(1.17s)	CPU time	21	25	115	1130	15000
49-cell	1.(5.07s)	CPU time	5	6	27	261	3550

Table 2: Model Comparisons

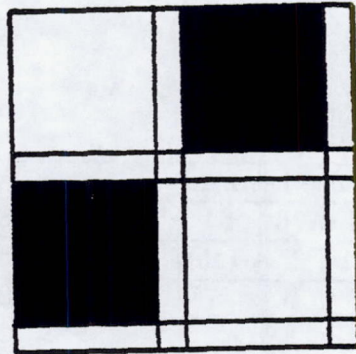
case	GMC	F.E. (272)	Continuum
4-cell	1.(1.17s)	1130	0.8
49-cell	1.(5.07s)	261	0.2



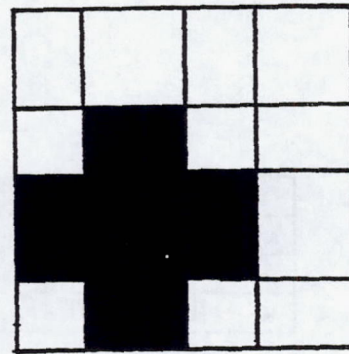
IDP = 1



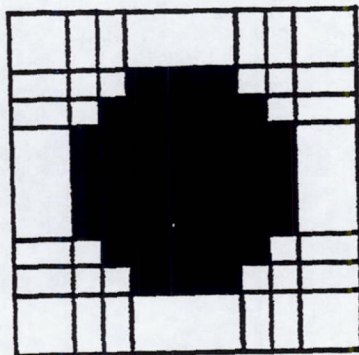
IDP = 2



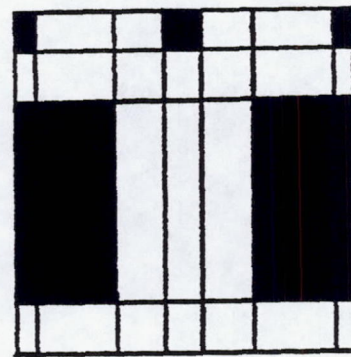
IDP = 3



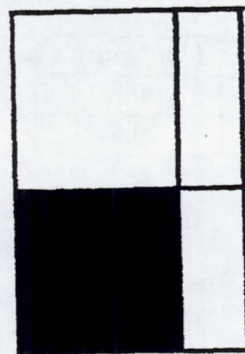
IDP = 4



IDP = 6



IDP = 9



IDP = 11

Matrix



Fiber



Figure 1: GMC Representative Volume Elements

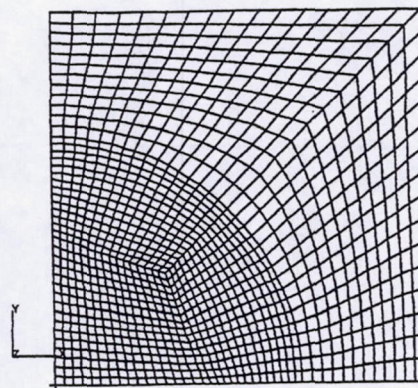
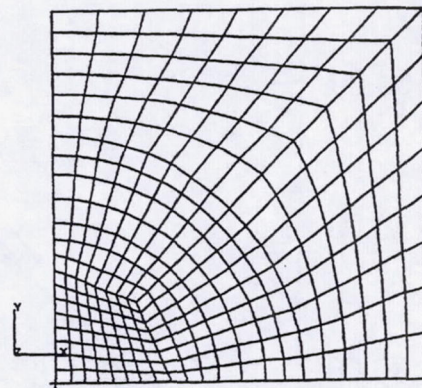
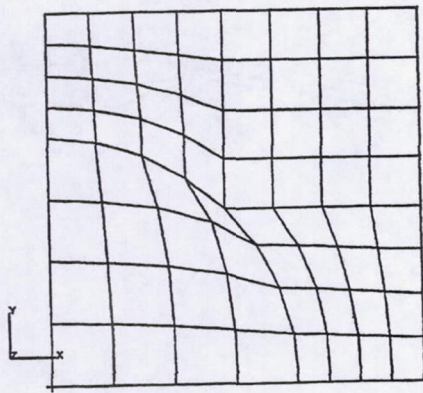
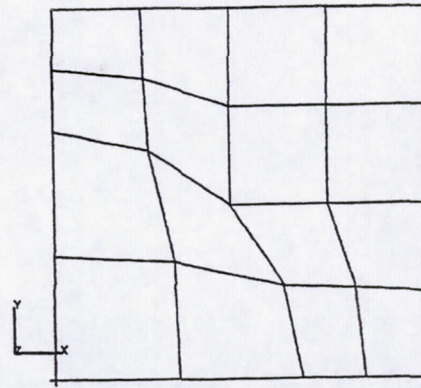
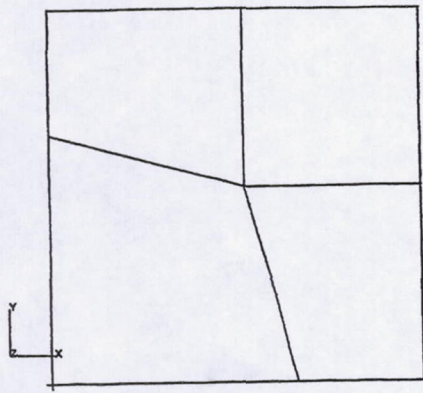


Figure 2: Unit Cell Finite Element Meshes

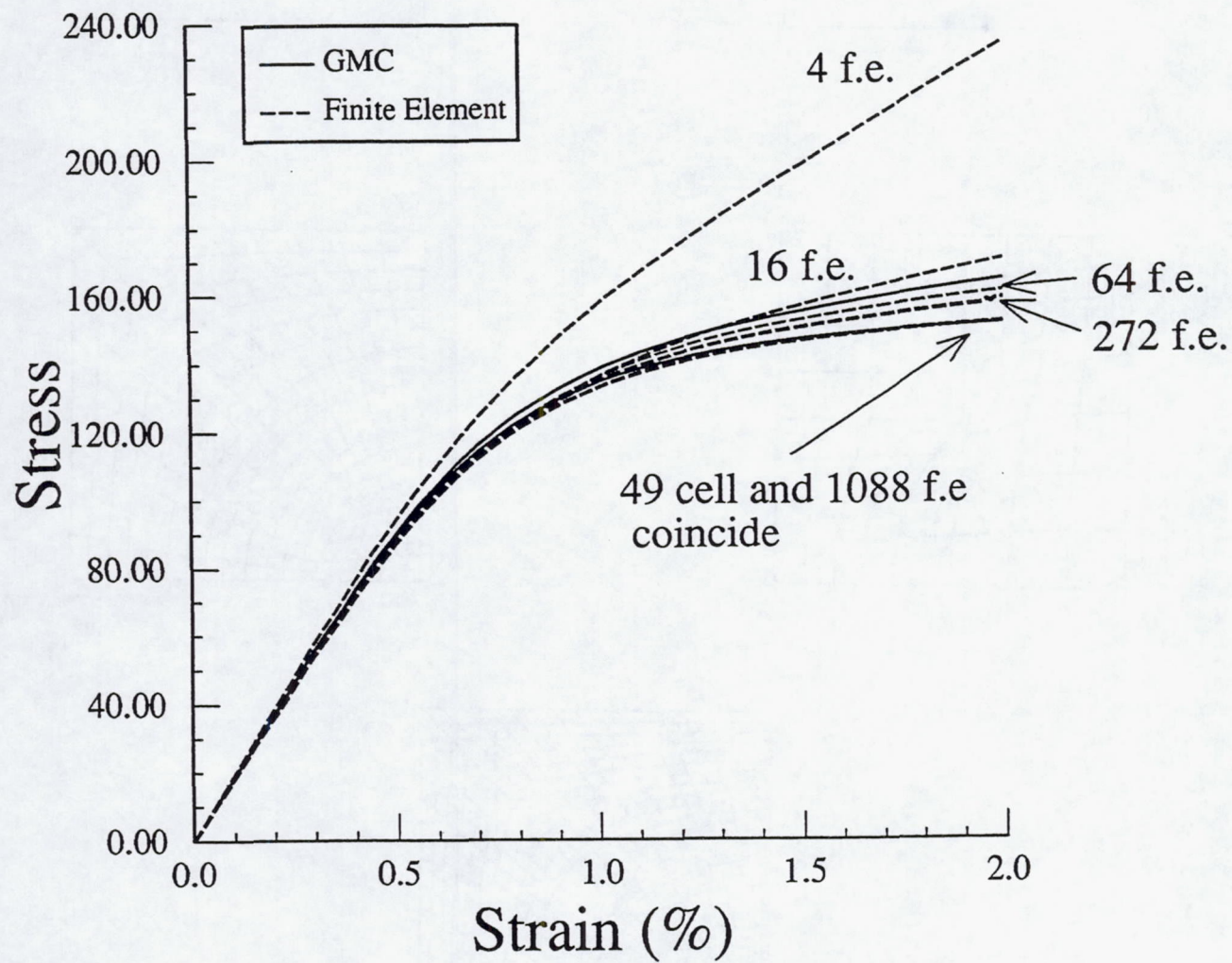


Figure 3: Transverse Deformation Comparison

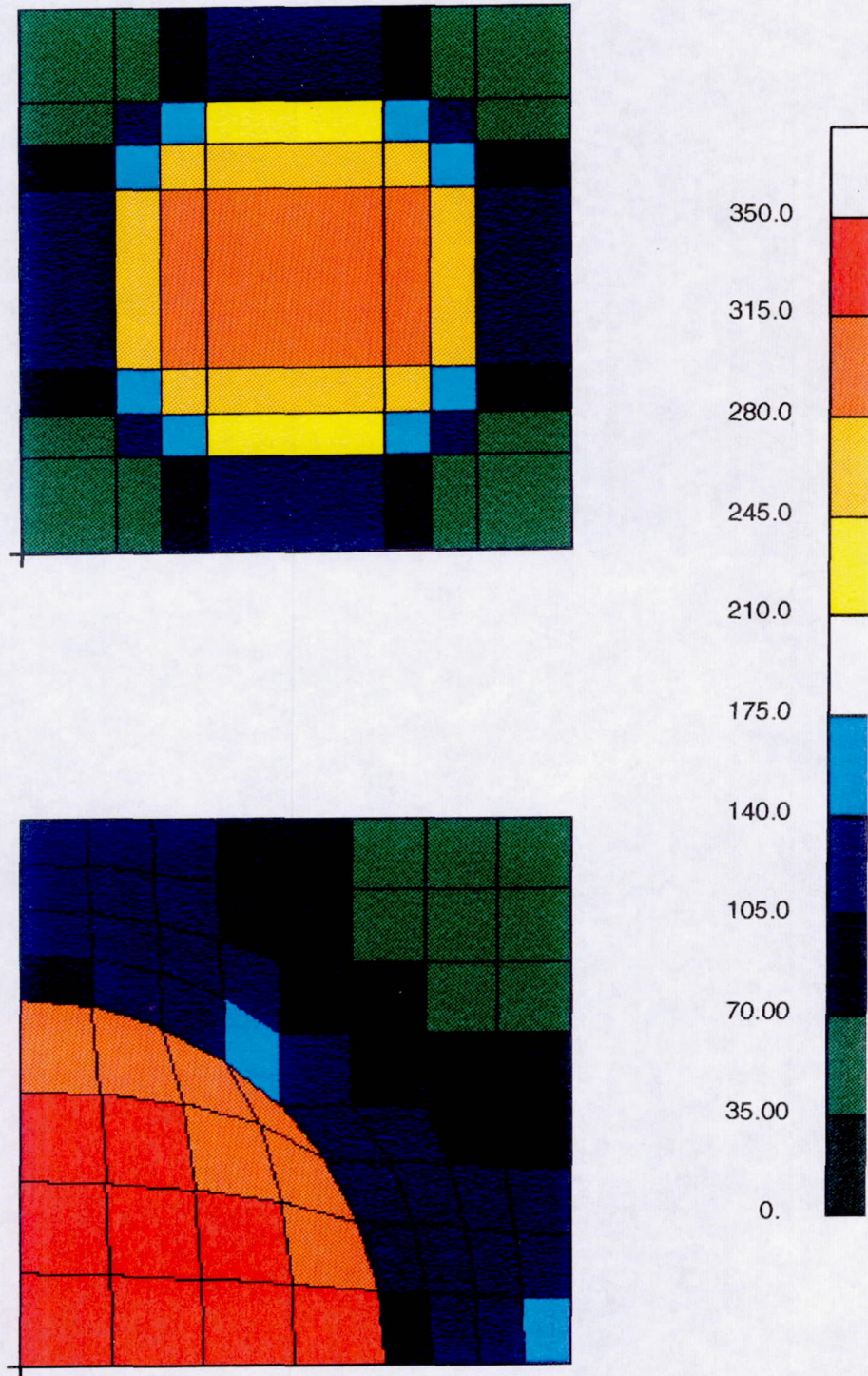
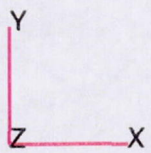


Figure 4: Effective Stress Distribution

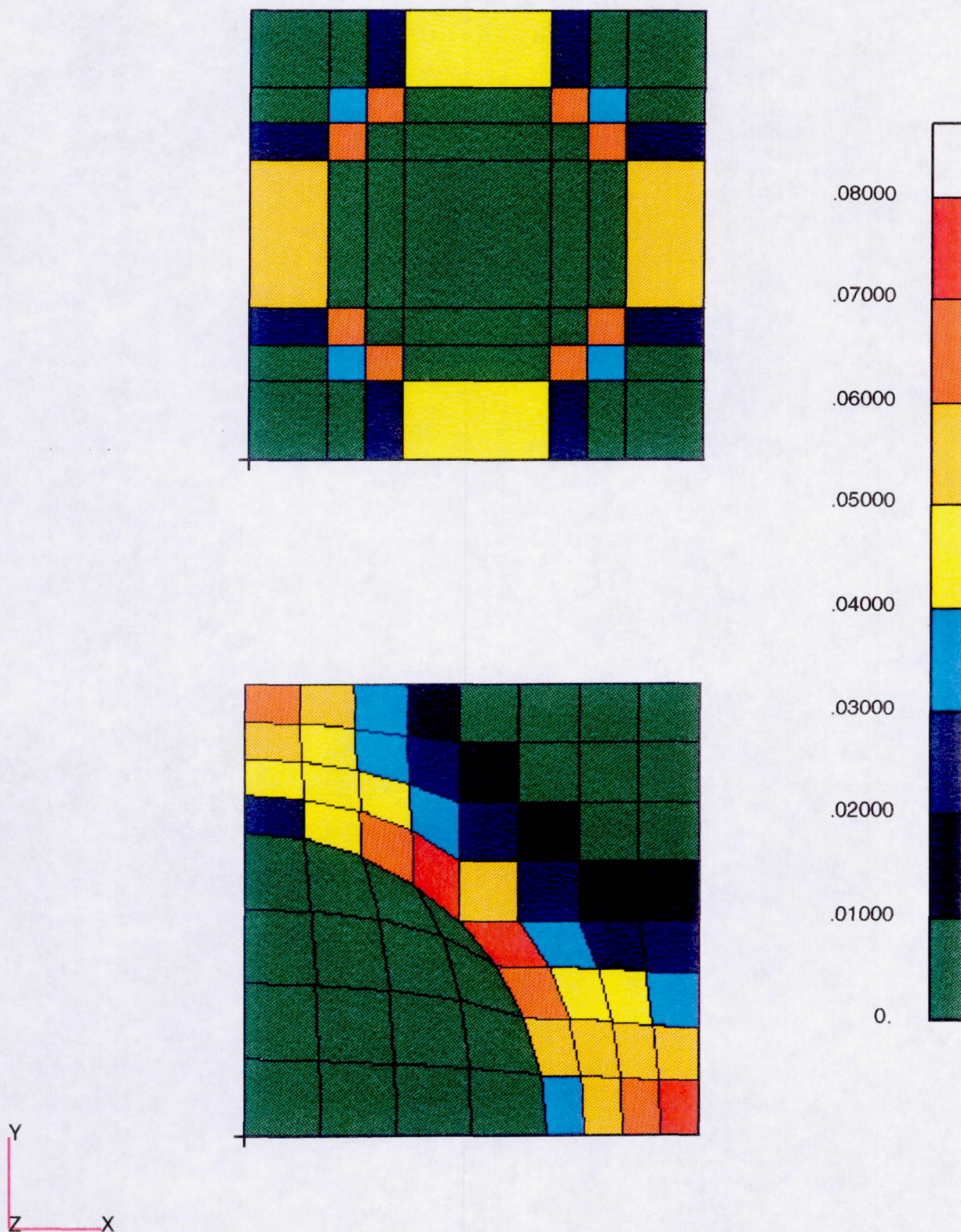


Figure 5: Effective Inelastic Strain Distribution

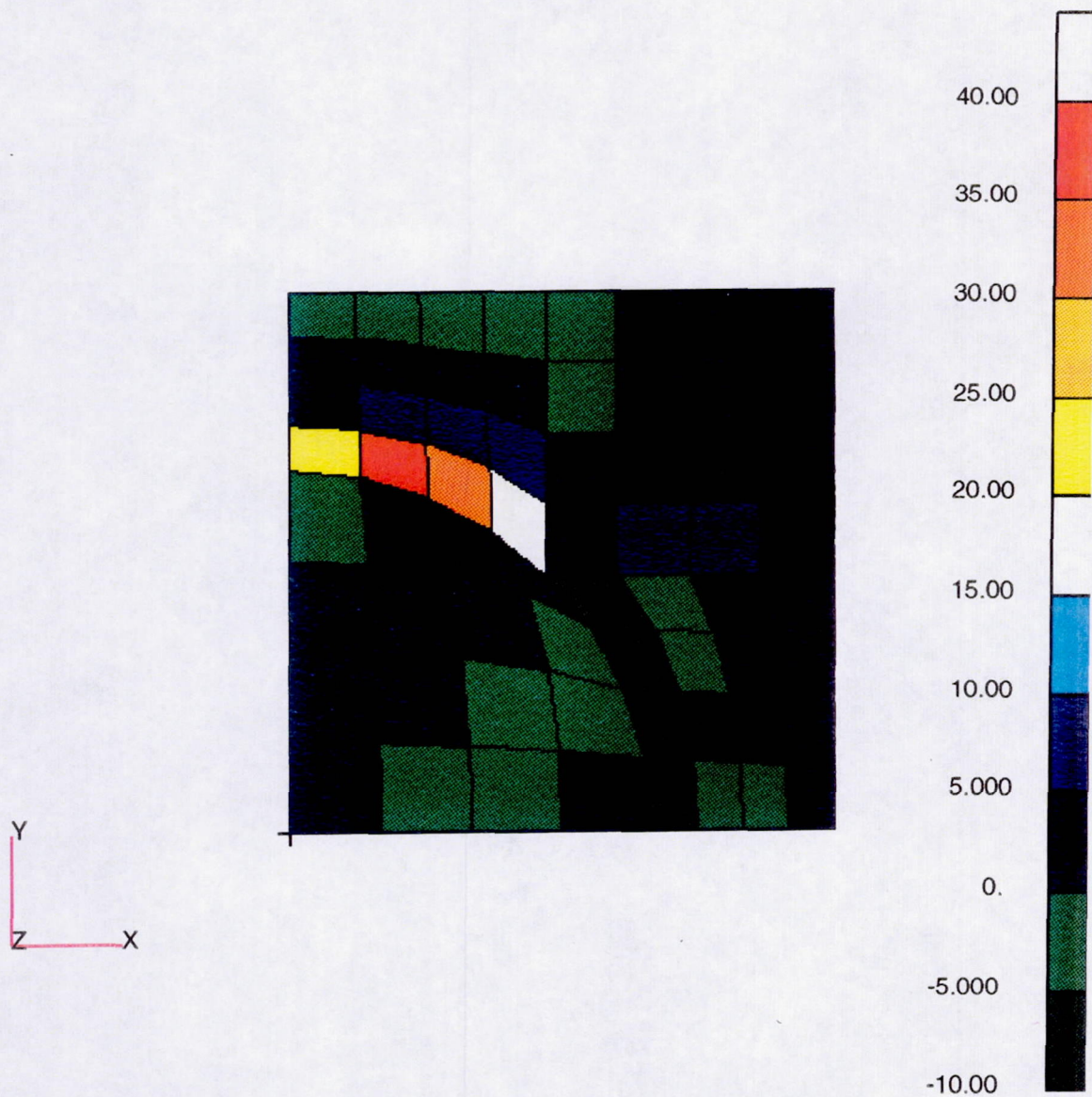


Figure 6: Finite Element Shear Stress Distribution

[illegible]


```

C
C
C   Purpose:  generalized cells model for metal matrix long-fiber
C             composites the fibers are oriented in the x1-direction
C
C             NBM   = MAX. NUMBER OF SUBCELLS IN BETA-DIRECTION (X2)
C             NGM   = MAX. NUMBER OF SUBCELLS IN GAMMA-DIRECTION (X3)
C             NMM   = MAX. NUMBER OF MATERIALS
C             NSM   = NBM*NGM = MAX. NUMBER OF SUBCELLS IN A
C                   REPRESENTATIVE CELL
C             CS    = CONSTITUENT ELASTIC STIFFNESS TENSOR
C             CG    = GLOBAL (EFFECTIVE) ELASTIC STIFFNESS TENSOR
C             CI    = INVERSE OF CG
C             SG    = GLOBAL STRESS TENSOR
C             EG    = GLOBAL STRAIN TENSOR
C             EPG   = GLOBAL PLASTIC STRAIN TENSOR
C             ETG   = GLOBAL THERMAL STRAIN TENSOR
C             SS    = SUBCELL STRESS TENSOR
C             ES    = SUBCELL STRAIN TENSOR
C             EPS   = SUBCELL PLASTIC STRAIN TENSOR
C             EPD   = SUBCELL PLASTIC STRAIN-RATE TENSOR
C             ETS   = SUBCELL THERMAL STRAIN TENSOR
C             DTEMP = DEVIATION FROM THE REFERENCE TEMPERATURE
C
C             NINDPT= FLAG WHEN MATERIAL CONSTANTS ARE INDEPENDENT
C                   OF TEMPERATURE
C
C             NOTE THAT EG(4)=2*EPS23 ,EG(5)=2*EPS13 , EG(6)=2*EPS12
C             AND SIMILAR RELATIONS HOLD FOR EPG(4),EPG(5),EPG(6)
C
C
C             Limit is currently set for a maximum of 4 subcells !!!!
C                                     ~~~~~
C
C             I/O FILES:
C                 UNIT | CONTENTS
C                 -----
C                 #60 - GRID INFO
C                 #61 - PATGEO.GMC

```



```

C          #62 - PATSTR.GMC
C          #66 - PROGRAM TRACE FILE
C=====
C
C-----
C Set parameters for generalized method of cells
C-----
C
C      NPLVL      = 1
C
C      FORDIS     = .TRUE.
C      HEAT       = .FALSE.
C      THME       = .FALSE.
C      NINDPT     = .TRUE.
C
C      IF(INC.EQ.0) THEN
C          ICOUNT = 1
C      ELSE
C          ICOUNT = ICOUNT + 1
C      ENDIF
C
C      IDP        = 1
C      NB         = 2
C      NG         = 2
C      PATRAN     = .FALSE.
C      NCMD       = 1
C      TEMPR      = 0.
C      DTEMPR     = 0.
C-----
C Assign subcell material id.
C-----
C      MATNUM(1,1) = 1
C      MATNUM(1,2) = 2
C      MATNUM(2,1) = 2
C      MATNUM(2,2) = 2
C      MATT(1)    = 1
C      MATT(2)    = 2
C
C-----

```



```

C  Material constants:
C  NMT - number of different materials
C  NST - number of subcells
C  VF  - fiber volume fraction
C  VC  - coating volume fraction
C-----
C
      NMT = 2
      NST = 4
      VF  = 0.35
      VC  = 0.
C
C-----
C  Elastic constituent material properties:
C
C  Fiber:
C-----
C
      EA(1)  = 58.D3
      ET(1)  = 58.D3
      GA(1)  = 23.2D3
      FNA(1) = 0.25
      FNT(1) = 0.25
      ALPA(1) = 1.
      ALPT(1) = 1.
C D_0
      VPROB(1,1) = 0.
C Z_0
      VPROB(1,2) = 1.
C Z_1
      VPROB(1,3) = 1.
C m
      VPROB(1,4) = 1.
C n
      VPROB(1,5) = 1.
C Q
      VPROB(1,6) = 1.
C
C-----

```



```

C Matrix
C-----
C
      EA(2)   = 11233.
      ET(2)   = 11233.
      GA(2)   = 4255.
      FNA(2)  = 0.32
      FNT(2)  = 0.32
      ALPA(2) = 1.
      ALPT(2) = 1.
C
      VPROB(2,1) = 10.
      VPROB(2,2) = 1600.
      VPROB(2,3) = 1600.
      VPROB(2,4) = 10.
      VPROB(2,5) = 0.2
      VPROB(2,6) = 10.
C=====
C
C-----
C Form elastic material stiffness
C-----
C
      IF((NCYCLE+INC) .EQ. 0) THEN
        GOTO 5100
      ELSEIF(NCYCLE .EQ. 0) THEN
        GOTO 5100
      ELSE
C
      ENDIF
C-----
C Set global time step: GDT
C 40 steps
C eps_max = 0.02 rate = 0.001
C-----
C
      GDT = 0.5
C
C-----

```



```

5100  CONTINUE
C
C-----
C  Square packing (original method of cells with 4 subcells)
C-----
C
      IF(IDP.EQ.1 ) THEN
        XH(1)=SQRT(VF)
        XH(2)=1-SQRT(VF)
        XL(1)=XH(1)
        XL(2)=XH(2)
C
C-----
C  Triangle packing
C-----
C
      ELSEIF(IDP .EQ. 2) THEN
        IF(VF.LT.0.288675) THEN
          XA=SQRT(SQRT(3.DO)*VF/2.DO)
          XB=(1-XA)/2.0
          XC=SQRT(3.DO/4.DO)-XA
          XH(1)=XB-XA/2
          XH(2)=XA
          XH(3)=XB-XA/2
          XH(4)=XA
          XL(1)=XA
          XL(2)=XC
          XL(3)=XA
          XL(4)=XC
        ELSE
          XA=DSQRT(DSQRT(3.DO)*VF/2.DO)
          XB=(1-XA)/2
          XC=DSQRT(3.DO/4.DO)-XA
          XH(1)=2.0*XB
          XH(2)=XA/2.0-XB
          XH(3)=2.0*XB
          XH(4)=XA/2.0-XB
          XL(1)=XA
          XL(2)=XC

```



```

        XL(3)=XA
        XL(4)=XC
    ENDIF

C
C-----
C  Square diagonal packing
C-----
C
    ELSEIF(IDP .EQ. 3) THEN
        XA=DSQRT(VF/2.DO)
        XH(1)=(1.-2.*XA)/2.
        XH(2)=XA
        XH(3)=(1-2*XA)/2
        XH(4)=XA
        XL(1)=XA
        XL(2)=(1-2*XA)/2
        XL(3)=XA
        XL(4)=(1-2*XA)/2

C
C-----
C  Cross shape fiber (square packing)
C-----
C
    ELSEIF(IDP .EQ. 4) THEN
        XB=-2.*XA+DSQRT(4.DO*XA**2.DO*VF)
        XH(1)=XA
        XH(2)=XB
        XH(3)=XA
        XH(4)=1-2*XA-XB
        XL(1)=XH(1)
        XL(2)=XH(2)
        XL(3)=XH(3)
        XL(4)=XH(4)

C
C-----
C  Coated fiber (square packing); vc=coating volume fraction
C-----
C
    ELSEIF(IDP .EQ. 5) THEN

```



```

XA=DSQRT(VF)
XH(2)=XA
XH(1)=(-XA+DSQRT(VF+VC))/2.
XH(3)=XH(1)
XH(4)=1-(XH(1)+XH(2)+XH(3))
XL(1)=XH(1)
XL(2)=XH(2)
XL(3)=XH(3)
XL(4)=XH(4)
C
C-----
C  Circular fiber approximation
C  (49 cell model)
C-----
C
ELSEIF(IDP .EQ. 6) THEN
  RADIUS=DSQRT(VF/PAI)
  HH=1.0
  XH(2)=DSQRT(PAI)*RADIUS/DSQRT(52.DO)
  XH(3)=XH(2)
  XH(4)=4.0*XH(2)
  XH(5)=XH(3)
  XH(6)=XH(2)
  XH(1)=(HH-2.0*XH(2)-2.0*XH(3)-XH(4))/2.0
  XH(7)=XH(1)
  XL(1)=XH(1)
  XL(2)=XH(2)
  XL(3)=XH(3)
  XL(4)=XH(4)
  XL(5)=XH(5)
  XL(6)=XH(6)
  XL(7)=XH(7)
ENDIF
C
C-----
C  Calculate subcell volume
C  {VS}
C-----
C

```



```

        SUMH=0
        DO 1010 I=1,NB
            SUMH=SUMH+XH(I)
1010    CONTINUE
        SUML=0
        DO 1012 I=1,NG
            SUML=SUML+XL(I)
1012    CONTINUE
C
        DO 1113 IB=1,NB
            DO 1114 IG=1,NG
                VS(NG*(IB-1)+IG) = XH(IB)*XL(IG)
1114    CONTINUE
1113    CONTINUE
C
C-----
C Subcell information output
C (optional)
C-----
C
        IF(NPLVL.GE.5) THEN
            WRITE(66,683)
            WRITE(66,684)
        ENDIF
C
        TOTV=0.0
        DO 1120 IB=1,NB
            DO 1120 IG=1,NG
                IS=NG*(IB-1)+IG
                MN=MATNUM(IB,IG)
                KK = MATT(MN)
                TOTV=TOTV + VS(IS)
                IF(NPLVL.GE.5) THEN
                    WRITE(66,685) IB,IG,IS,MN,KK,VS(IS)
                ENDIF
1120    CONTINUE
C
C-----
C Calculate subcell material volume ratio

```



```

C  {VSMR}
C-----
C
      DO 1130 MN=1,NMT
        VSMR(MN)=0
        DO 1130 IS=1,NB*NG
          VS1(IS,MN)=0
1130    CONTINUE
      DO 1140 IB=1,NB
        DO 1140 IG=1,NG
          IS=NG*(IB-1)+IG
          MN=MATNUM(IB,IG)
          VS1(IS,MN)=VS(IS)
1140    CONTINUE
      DO 1141 MN=1,NMT
        DO 1141 IS=1,NB*NG
          VSMR(MN)=VSMR(MN)+VS1(IS,MN)
1141    CONTINUE
      DO 1142 MN=1,NMT
        VSMR(MN)=VSMR(MN)/TOTV
        IF(NPLVL.GE.5) WRITE(66,143) MN,VSMR(MN)
1142    CONTINUE
C
C-----
C  Initialization
C-----
C
      NST=NB*NG
      DO 1150 I=1,6
        DO 1150 J=1,6
          DO 1150 K=1,NMM
            CS(I,J,K)=0
1150    CONTINUE
      DO 1160 I=1,6
        DO 1160 J=1,6
          CI(I,J)=0
          CG(I,J)=0
1160    CONTINUE
C

```



```

C-----
C  DEFINE THE TEMPERATURE-DEPENDENT PROPERTIES AT 6 DIFFERENT
C  TEMPERATURES: THRM(I) ,I=1...6
C  FOR INTERMEDIATE TEMPERATURES APPLY A LINEAR INTERPOLATION
C  ==> CURRENTLY ONLY FOR BODNER MODEL <==
C-----
C
      IF(HEAT .OR. THME) THEN
        IF(NCMD.EQ.1) THEN
          TEMPR = CTEMP
          IF(TEMPR.GE.THRM(1).AND.TEMPR.LE.THRM(2)) THEN
            I=2
          ELSEIF(TEMPR.GE.THRM(2).AND.TEMPR.LE.THRM(3)) THEN
            I=3
          ELSEIF(TEMPR.GE.THRM(3).AND.TEMPR.LE.THRM(4)) THEN
            I=4
          ELSEIF(TEMPR.GE.THRM(4).AND.TEMPR.LE.THRM(5)) THEN
            I=5
          ELSEIF(TEMPR.GE.THRM(5).AND.TEMPR.LE.THRM(6)) THEN
            I=6
          ELSE
            WRITE(66,*) 'GMC: **** ERROR ****'
            WRITE(66,*) 'TEMPERATURE IS OUT OF RANGE ',TEMPR
            STOP
          ENDIF
        DIFF=(TEMPR-THRM(I))/(THRM(I)-THRM(I-1))
        DO 1200 KK=1,NMT
          IF(NPLVL.GE.5) WRITE(66,*) 'GMC:  NM ',NM,' KK ',KK
          EA(KK)  = EATM(I,KK)+
          &        (EATM(I,KK)-EATM(I-1,KK))*DIFF
          FNA(KK) = FNATM(I,KK)+
          &        (FNATM(I,KK)-FNATM(I-1,KK))*DIFF
          ET(KK)  = ETTM(I,KK)+
          &        (ETTM(I,KK)-ETTM(I-1,KK))*DIFF
          FNT(KK) = FNTTM(I,KK)+
          &        (FNTTM(I,KK)-FNTTM(I-1,KK))*DIFF
          GA(KK)  = GATM(I,KK)+
          &        (GATM(I,KK)-GATM(I-1,KK))*DIFF
          ALPA(KK) = ALPATM(I,KK)+

```



```

c      &                (ALPATM(I,KK)-ALPATM(I-1,KK))*DIFF
c      ALPT(KK) = ALPTTM(I,KK)+
c      &                (ALPTTM(I,KK)-ALPTTM(I-1,KK))*DIFF
c1200      CONTINUE
C
      ELSEIF(NCMD.NE.1) THEN
c      CALL CONEVL(
c      DO 1210 KK=1,NMT
c      EA(KK) = PEM(1,KK)
c      ET(KK) = PEM(2,KK)
c      GA(KK) = PEM(3,KK)
c      FNA(KK) = PEM(4,KK)
c      FNT(KK) = PEM(5,KK)
c      ALPA(KK) = PEM(6,KK)
c      ALPT(KK) = PEM(7,KK)
c1210      CONTINUE
      ELSE
        WRITE(66,*) 'GMC: ERROR IN TEMPERATURE',
&                ' DEPENDENT CONSTANTS'
        STOP
      ENDIF
    ENDIF
  C
  C-----
  C Elastic stiffness matrix [CS] for each subcell
  C-----
  C
    DO 1230 NM=1,NMT
      GT = 0.5*ET(NM)/(1+FNT(NM))
      FK = 0.25*EA(NM)/(0.5*(1-FNT(NM))*
&      (EA(NM)/ET(NM))-FNA(NM)**2)
      CS(1,1,NM) = EA(NM) + 4*FK*FNA(NM)**2
      CS(2,1,NM) = 2*FK*FNA(NM)
      CS(3,1,NM) = 2*FK*FNA(NM)
      CS(1,2,NM) = 2*FK*FNA(NM)
      CS(2,2,NM) = FK+GT
      CS(3,2,NM) = FK-GT
      CS(1,3,NM) = 2*FK*FNA(NM)
      CS(2,3,NM) = FK-GT

```



```

        CS(3,3,NM) = FK+GT
        CS(4,4,NM) = GT
        CS(5,5,NM) = GA(NM)
        CS(6,6,NM) = GA(NM)
C
        IF(NPLVL.GE.5) THEN
c          IF(NPLVL.GE.2 .and. inc.ge.0 .and. nn.eq.1) then
write(66,*) 'nm ',nm
write(66,*) 'ns ',ns
        WRITE(66,8266)
        WRITE(66,8267) CS(1,1,NM),CS(1,2,NM),CS(1,3,NM)
        WRITE(66,8267) CS(2,1,NM),CS(2,2,NM),CS(2,3,NM)
        WRITE(66,8267) CS(3,1,NM),CS(3,2,NM),CS(3,3,NM)
        WRITE(66,8268) CS(4,4,NM),CS(5,5,NM),CS(6,6,NM)
        ENDIF
C
C-----
C Thermal stress vectors [GAM] of subcells
C-----
C
        GAM(1,NM)=CS(1,1,NM)*ALPA(NM)+(CS(1,2,NM)+
&          CS(1,3,NM))*ALPT(NM)
        GAM(2,NM)=CS(1,2,NM)*ALPA(NM)+(CS(2,2,NM)+
&          CS(2,3,NM))*ALPT(NM)
        GAM(3,NM)=CS(1,3,NM)*ALPA(NM)+(CS(2,3,NM)+
&          CS(3,3,NM))*ALPT(NM)
1230    CONTINUE
C
C-----
C Compute the concentration factor
C initialization of the a-matrix
C-----
C
        DO 1240 J=1, 6*NST + 6 + 6*NST
            DO 1250 I=1,6*NST
                A(I,J)=0.000
1250    CONTINUE
1240    CONTINUE
C

```



```

      NSR = NG*(IB-1) + IGR
      MSL = MATNUM(IB,IGL)
      MSR = MATNUM(IB,IGR)
      NA=NA + 1
      DO 1650 NC=1,6
        A(NA,6*(NSL-1)+NC) = CS(MC,NC,MSL)
        A(NA,6*(NSR-1)+NC) = -CS(MC,NC,MSR)
        A(NA,6*(NST+NSL)+NC) = CS(MC,NC,MSL)
        A(NA,6*(NST+NSR)+NC) = -CS(MC,NC,MSR)
1650      CONTINUE
1660      CONTINUE
1670      CONTINUE
1680      CONTINUE
C
C-----
C Displacement continuity conditions
C-----
C-----
C ES11(B,G)=EG11   B=1,...,NB ; G=1,...,NG
C-----
C
      DO 1710 IB=1,NB
        DO 1705 IG=1,NG
          NA=NA+1
          A(NA,6*NB*NG+1) = 1
          NS = NG*(IB-1)+IG
          A(NA,6*(NS-1)+1) = 1
1705      CONTINUE
1710      CONTINUE
C
C-----
C SUM [H(B)*ES22(B,G)]=H*EG22      G=1,...,NG
C-----
C
      DO 1720 IG=1,NG
        NA=NA+1
        A(NA,6*NB*NG+2) = SUMH
        DO 1715 IB=1,NB
          NS = NG*(IB-1)+IG

```



```

          A(NA,6*(NS-1)+2) = XH(IB)
1715      CONTINUE
C
C-----
C  SUM [H(B)*ES12(B,G)]=H*EG12      G=1,...,NG
C-----
C
          NA=NA+1
          A(NA,6*NB*NG+6) = SUMH
          DO 1718 IB=1,NB
              NS = NG*(IB-1)+IG
              A(NA,6*(NS-1)+6) = XH(IB)
1718      CONTINUE
1720      CONTINUE
C
C-----
C  SUM [L(G)*ES33(B,G)]=L*EG33      B=1,...,NB
C-----
C
          DO 1730 IB=1,NB
              NA=NA+1
              A(NA,6*NB*NG+3) = SUML
              DO 1725 IG=1,NG
                  NS = NG*(IB-1)+IG
                  A(NA,6*(NS-1)+3) = XL(IG)
1725      CONTINUE
C
C-----
C  SUM [L(G)*ES13(B,G)]=L*EG13      B=1,...,NB
C-----
C
          NA=NA+1
          A(NA,6*NB*NG+5) = SUML
          DO 1728 IG=1,NG
              NS = NG*(IB-1)+IG
              A(NA,6*(NS-1)+5) = XL(IG)
1728      CONTINUE
1730      CONTINUE
C

```



```

C-----
C  SUM SUM [H(B)*L(G)*ES23(B,G)]=H*L*EG23
C-----
C
      NA=NA+1
      DO 1740 IB=1,NB
        DO 1735 IG=1,NG
          A(NA,6*NB*NG+4) = SUMH*SUML
          NS = NG*(IB-1)+IG
          A(NA,6*(NS-1)+4) = XH(IB)*XL(IG)
1735      CONTINUE
1740      CONTINUE
C
      CALL SGEFA(A,NAM,6*NB*NG,IPVT,INFO)
      IF(INFO.NE.0) STOP
      DO 2110 I=1,6+6*NB*NG
        CALL SGESL(A,NAM,6*NB*NG,IPVT,A(1,6*NB*NG+I),0)
2110      CONTINUE
C
C-----
C  Compute the effective elastic stiffness matrix [CG]
C-----
C
      DO 2800 I=1,6
        DO 2790 J=1,6
          DO 2780 IB=1,NB
            DO 2770 IG=1,NG
              IS=NG*(IB-1)+IG
              IM=MATNUM(IB,IG)
              TMP=0.000
              DO 2760 M=1,6
                TMP=TMP + CS(I,M,IM)*A(6*(IS-1)+M,6*NS+J)
2760          CONTINUE
              CG(I,J)=CG(I,J) + VS(IS)*TMP/(SUMH*SUML)
2770          CONTINUE
2780          CONTINUE
2790          CONTINUE
2800          CONTINUE
C

```



```

C-----
C  MARC control statements
C-----
      IF((NCYCLE+INC) .EQ. 0) THEN
        CALL PLACE(CG,D,36)
        TEMP(1) = 0.
        TEMP(2) = 0.
        CALL ZEROR(SIGAL,181)
        CALL ZEROR(DSIGAL,181)
        CALL ZEROR(SIGL,36)
        CALL ZEROR(DSIGL,36)
        RETURN
      ELSEIF(NCYCLE .EQ. 0) THEN
        CALL PLACE(CG,D,36)
        RETURN
      ELSE
        CALL PLACE(CG,D,36)
        CALL PLACE(CG,CD,36)
      ENDIF

C
C-----
C  Invert the effective elastic stiffness matrix [CG]
C-----
C
      DET=CG(1,1)*(CG(2,2)*CG(3,3)-CG(2,3)**2)
&      -CG(1,2)*(CG(1,2)*CG(3,3)-CG(2,3)*CG(1,3))
&      +CG(1,3)*(CG(1,2)*CG(2,3)-CG(2,2)*CG(1,3))
      CI(1,1)=(CG(2,2)*CG(3,3)-CG(2,3)**2)/DET
      CI(1,2)=(-CG(1,2)*CG(3,3)+CG(1,3)*CG(2,3))/DET
      CI(1,3)=(CG(1,2)*CG(2,3)-CG(1,3)*CG(2,2))/DET
      CI(2,2)=(CG(1,1)*CG(3,3)-CG(1,3)*CG(1,3))/DET
      CI(2,3)=(-CG(1,1)*CG(2,3)+CG(1,2)*CG(1,3))/DET
      CI(3,3)=(CG(1,1)*CG(2,2)-CG(1,2)*CG(1,2))/DET
      CI(4,4)=1/CG(4,4)
      CI(5,5)=1/CG(5,5)
      CI(6,6)=1/CG(6,6)
      CI(2,1)=CI(1,2)
      CI(3,1)=CI(1,3)
      CI(3,2)=CI(2,3)

```



```

C
  IF(NPLVL.GE.5) THEN
    WRITE(66,869)
    WRITE(66,91) 1./CI(1,1)
    WRITE(66,92) -CI(1,2)/CI(1,1)
    WRITE(66,93) 1./CI(2,2)
    WRITE(66,94) -CI(2,3)/CI(2,2)
    WRITE(66,95) 1./CI(3,3)
    WRITE(66,96) 1./CI(4,4)
    WRITE(66,97) 1./CI(5,5)
    WRITE(66,98) 1./CI(6,6)
  ENDIF

C
C-----
C  Compute the effective coefficients
C-----
C
  DO 2801 I=1,3
    GAMS(I)=0.
  DO 2801 IG=1,NG
  DO 2801 IB=1,NB
    IS=NG*(IB-1)+IG
    MN=MATNUM(IB,IG)
    SUM=0
    DO 2761 K=1,3
      SUM=SUM +A(6*(IS-1)+K,6*NST+I)*GAM(K,MN)
2761  CONTINUE
      GAMS(I)=GAMS(I)+VS(IS)*SUM/(SUMH*SUML)
2801  CONTINUE
C
  ALPS(1)=CI(1,1)*GAMS(1)+CI(1,2)*GAMS(2)+CI(1,3)*GAMS(3)
  ALPS(2)=CI(2,1)*GAMS(1)+CI(2,2)*GAMS(2)+CI(2,3)*GAMS(3)
  ALPS(3)=CI(3,1)*GAMS(1)+CI(3,2)*GAMS(2)+CI(3,3)*GAMS(3)

C
C-----
C  Determine the effective cte directly from the model
C-----
C
  IF(HEAT .OR. THME) THEN

```



```

DO 3130 IB=1,NB
DO 3130 IG=1,NG
  NS=NG*(IB-1)+IG
  NM=MATNUM(IB,IG)
  KK = NM
  ETS(6*(NS-1)+1)=ALPA(KK)
  ETS(6*(NS-1)+2)=ALPT(KK)
  ETS(6*(NS-1)+3)=ALPT(KK)
  ETS(6*(NS-1)+4)=0
  ETS(6*(NS-1)+5)=0
  ETS(6*(NS-1)+6)=0
C
  DO 3140 JJ=1,6
    DSIGAL(37+(36*(NS-1))+(JJ+30))=ETS(6*(NS-1)+JJ)*DTEMPR
3140    CONTINUE
3130    CONTINUE
C
C-----
C  [UT]=[AP(B,G)]*{ETS}
C-----
C
  DO 3145 NS=1,NST
  DO 3145 I=1,6
    UT(I,NS)=0
    DO 3145 J=1,6*NST
      UT(I,NS)=UT(I,NS)+A(6*(NS-1)+I,6*NST+6+J)*ETS(J)
3145    CONTINUE
C
C-----
C  {HT}=SUM SUM (VS(B,G)*[CS]*([UT(B,G)]-[ETS(B,G)]))/TOTV
C-----
C
  DO 3150 I=1,3
  HT(I)=0
  DO 3150 IB=1,NB
  DO 3150 IG=1,NG
    NS=NG*(IB-1)+IG
    NM=MATNUM(IB,IG)
  DO 3150 M=1,3

```



```

        HT(I)=HT(I)+VS(NS)*
        &      CS(I,M,NM)*(UT(M,NS)-ETS(6*(NS-1)+M))/(SUMH*SUML)
3150    CONTINUE
C
C-----
C  {ALPS}=-[CI]*{HT}
C-----
C
        DO 3160 I=1,3
            ALPS(I) = 0.
        DO 3160 M=1,3
            ALPS(I) = ALPS(I)-CI(I,M)*HT(M)
3160    CONTINUE
C
C-----
C  CALCULATE INSTANTANEOUS THERMAL EXPANSION
C-----
C
C      CALL ALPEVL(ALPS,VS,SUMH,SUML,ALPSI)
      ELSE
        CALL ZEROR(ETS,NAM)
      ENDIF
C
C-----
C  Determination of hill concentration
C  tensors [HILL] in subcell (B,G)
C  [HILL(B,G)]=[CS(B,G)]*[A(B,G)]*[CI]
C-----
C
        DO 3170 IB=1,NB
        DO 3170 IG=1,NG
            NS=NG*(IB-1)+IG
            NM=MATNUM(IB,IG)
            DO 3170 I=1,6
            DO 3170 J=1,6
                HILL(NS,I,J)=0
            DO 3170 M=1,6
            DO 3170 N=1,6
                HILL(NS,I,J)=HILL(NS,I,J)+

```



```

&                                CS(I,M,NM)*A(6*(NS-1)+M,6*NST+N)*CI(N,J)
3170 CONTINUE
C
    IF(INC.ge.0 .AND. NPLVL.GE.5) THEN
        WRITE(6,7129)
        DO 3180 IB=1,NB
            DO 3180 IG=1,NG
                NS=NG*(IB-1)+IG
                WRITE(6,7126) IB,IG
                DO 3181 I=1,6
                    WRITE(66,7128) (HILL(NS,I,J),J=1,6)
3181          CONTINUE
3180          CONTINUE
        ENDIF
C
C-----
C  Determination of hill thermal concentration
C  vector [HILLT] in the subcells
C  [HILLT(B,G)]=-[CS(B,G)]*([UT(B,G)]-[ETS(B,G)]+[A(B,G)]*{ALPS})
C-----
C
    IF(HEAT .OR. THME) THEN
        DO 3185 I=1,3
            DO 3185 IB=1,NB
                DO 3185 IG=1,NG
                    NS=NG*(IB-1)+IG
                    NM=MATNUM(IB,IG)
                    HILLT(NS,I)=0
                    DO 3185 M=1,3
                        HT(M)=0
                        DO 3190 N=1,3
                            HT(M)=HT(M)+A(6*(NS-1)+M,6*NST+N)*ALPS(N)
3190          CONTINUE
                            HILLT(NS,I)=HILLT(NS,I)
&                                -CS(I,M,NM)*(UT(M,NS)-ETS(6*(NS-1)+M)+HT(M))
3185          CONTINUE
C
        IF(INC.EQ.0 .AND. NPLVL.GE.5) THEN
            WRITE(66,7139)

```



```

        DO 3200 IB=1,NB
        DO 3200 IG=1,NG
            NS=NG*(IB-1)+IG
            WRITE(66,7138) IB,IG,(HILLT(NS,I),I=1,3)
3200      CONTINUE
        ENDIF
    ENDIF
C
C-----
C  Finish elastic calculations
C-----
C  Begin inelastic calculations
C-----
C
        DO 8800 IB=1,NB
        DO 8750 IG=1,NG
            NS = NG*(IB-1) + IG
            NM = MATNUM(IB,IG)
C
            DO 2222 J=1,6
                EPSOSC(J) = TEMP((2 + 37 + (36*(NS-1))) +J)
                STROSC(J) = TEMP((2 + 37 + (36*(NS-1))) +J+6)
                EPIOSC(J) = TEMP((2 + 37 + (36*(NS-1))) +J+12)
2222      CONTINUE
C
C-----
C  Copy subcell quantities from
C  {SIGAL} TO {SIGL} AND
C  {DSIGAL} TO {DSIGL}
C-----
C
        DO 7705 KK=1,36
            SIGL(KK) = SIGAL(37+(36*(NS-1))+KK)
            DSIGL(KK) = DSIGAL(37+(36*(NS-1))+KK)
7705      CONTINUE
C
C-----
C  Call constitutive model to get:
C  inelastic strain rate {EPS}

```



```

C and state variable rates
C-----
C
      IF(NCMD .EQ. 1) THEN
        CALL BODNER(DSIGL,SIGL,STROSC,NM,NS,INC,N,NN)
      ELSE
C
C Insert other viscoplastic material models here
C
        WRITE(66,*) 'ILLEGAL MODEL'
        STOP
      ENDIF
C
C-----
C Copy subcell quantities into appropriate
C postions in {SIGAL} and {DSIGAL}
C-----
C
      DO 7700 KK=1,36
        SIGAL(37+(36*(NS-1))+KK) = SIGL(KK)
        DSIGAL(37+(36*(NS-1))+KK) = DSIGL(KK)
7700    CONTINUE
C
C-----
C Copy subcell plastic strain rate {EPS}
C from {DSIGAL}
C-----
C
      EPS(6*(NS-1)+1)=DSIGAL(37+(36*(NS-1)+13))
      EPS(6*(NS-1)+2)=DSIGAL(37+(36*(NS-1)+14))
      EPS(6*(NS-1)+3)=DSIGAL(37+(36*(NS-1)+15))
      EPS(6*(NS-1)+4)=DSIGAL(37+(36*(NS-1)+16))
      EPS(6*(NS-1)+5)=DSIGAL(37+(36*(NS-1)+17))
      EPS(6*(NS-1)+6)=DSIGAL(37+(36*(NS-1)+18))
8750  CONTINUE
8800  CONTINUE
C-----
C End subcell loop
C-----

```



```

C
C-----
C  Compute the average composite plastic strain rate {EPG}
C-----
C
      DO 8930 IS=1,NST
        DO 8920 I=1,6
          U(I,IS)=0.000
          DO 8910 J=1,6*NST
            U(I,IS)=U(I,IS) + A(6*(IS-1)+I,6*NST+6+J)*EPS(J)
8910      CONTINUE
8920      CONTINUE
8930      CONTINUE
C
      DO 8950 I=1,6
        H(I)=0.000
        DO 8960 IB=1,NB
          DO 8960 IG=1,NG
            IS=NG*(IB-1)+IG
            IM=MATNUM(IB,IG)
            DO 8940 J=1,6
              H(I)=H(I) +VS(IS)*CS(I,J,IM)*( U(J,IS)-EPS(6*(IS-1)+J) )
&              /(SUMH*SUML)
8940      CONTINUE
8960      CONTINUE
8950      CONTINUE
C
      DO 8980 I=1,6
        EPG(I)=0.000
        DO 8970 J=1,6
          EPG(I) = EPG(I) - CI(I,J)*H(J)
8970      CONTINUE
        DSIGAL(I+12) = EPG(I)
8980      CONTINUE
C
C-----
C  Global strains {DSIGAL(1-6)}
C-----
C-----

```



```

C Subcell loop
C-----
C
      DO 9775 IB=1,NB
      DO 9775 IG=1,NG
C
C-----
C Compute the average strain rate {ES} in the subcell
C {ES}= [A(B,G)]*{EG} + [AP(B,G)]*{EPS} + [AP(B,G)]*{ETS}
C-----
C
      NS = NG*(IB-1) + IG
      NM = MATNUM(IB,IG)
      DO 9150 M=1,6
        ES(M) = U(M,NS) + UT(M,NS)*DTEMPR
        DO 9100 N=1,6
          ES(M) = ES(M) + A(6*(NS-1)+M,6*NST+N)*EG(N)
9100      CONTINUE
          DSIGAL(37+(36*(NS-1))+M) = ES(M)
9150      CONTINUE
C
C-----
C Compute the average stress rate {SS} in the subcell
C {SS}=[CS]*(ES(B,G)-EPS(B,G)-ETS(B,G))
C-----
C
      CALL ZEROR(SS,6)
      DO 9250 M2=1,6
        DO 9200 N2=1,6
          SS(M2) = SS(M2) + CS(M2,N2,NM)*(ES(N2)-EPS(6*(NS-1)+N2)-
&          ETS(6*(NS-1)+N2)*DTEMPR)
9200      CONTINUE
C
C
C
      IF(NINDPT) GOTO 9010
C
C-----
C Add additional term in stress rate equation

```



```

C  when [CS] is a function of temperature
C-----
C
      IF(HEAT .OR. THME) THEN
        IF(M2.EQ.1) THEN
c          CALL CSEVAL(CPDRV)
        ENDIF
        IF(M2.EQ.1) THEN
          DO 9701 I=1,6
            ESO(I) = SIGAL(37+(36*(NS-1))+I)
            EPSO(I) = SIGAL(37+(36*(NS-1)+(I+12)))
            ETSO(I) = SIGAL(37+(36*(NS-1)+(I+30)))
            TEMPO = SIGAL(37)
9701      CONTINUE
          ENDIF
          DO 9251 N3=1,6
            ADDTRM = CPDRV(M,N3)*DTEMPR*(ESO(N3)-EPSO(N3)-ETSO(N3))
            SS(M2) = SS(M2) +
              &      CPDRV(M2,N3)*DTEMPR*(ESO(N3)-EPSO(N3)-ETSO(N3))
9251      CONTINUE
          ENDIF
C
9010      CONTINUE
C
C-----
C  Copy subcell stress rate {SS} to {DSIGAL}
C-----
C
      DSIGAL(37+(36*(NS-1))+(M2+6)) = SS(M2)
9250      CONTINUE
9775      CONTINUE
C
C-----
C  Compute the average composite
C  thermal strain rates {ETG}
C  {ALPSI} - instantaneous thermal
C            expansion coefficient
C-----
C

```



```

CALL ZEROR(ETG,6)
IF(HEAT .OR. THME) THEN
    ETG(1) = ALPSI(1)*DTEMPR
    ETG(2) = ALPSI(2)*DTEMPR
    ETG(3) = ALPSI(3)*DTEMPR
    ETG(4) = 0
    ETG(5) = 0
    ETG(6) = 0
ENDIF
C
C-----
C   Copy global thermal strain rates
C   to {DSIGAL(31-36)}
C-----
C
    DO 9889 JJ=1,6
c       dsigal(jj+30) = etg(jj)
        DSIGAL(JJ+30) = 0.0
9889 CONTINUE
C
C-----
C   Compute the average composite stress rate field {SG}
C   and copy to {DSIGAL(7-12)}
C-----
C
    DO 9890 I=1,6
        SG(I)= 0.000
        DO 9885 J=1,6
            SG(I) = SG(I) + CG(I,J)*(EG(J)-EPG(J)-ETG(J))
9885 CONTINUE
        DSIGAL(I+6) = SG(I)
9890 CONTINUE
C
C   /\ /\ /\ /\ /\ /\ /\ /\ /\ /\ /\ /\ /\ /\ /\ /\ /\ /\ /\ /\
C
C   End generalized method of cells code
C=====
C
C-----

```



```

C  Integrate rates using forward euler
C-----
C
C-----
C  Form MARC increment vector {DTEMP}
C-----
C
      LIM = 37 + NST*36
      DO 5000 K = 1,LIM
        DTEMP(K+2) = GDT*DSIGAL(K)
5000  CONTINUE
C
C-----
C  {G} - MARC vector of inelastic strain increment
C-----
C
      CALL ZEROR(G,6)
      DO 5010 I = 1,6
        DO 5010 J = 1,6
          G(I) = G(I) + GDT*CD(I,J)*DSIGAL(J+12)
5010  CONTINUE
C
C-----
C  Update MARC stress vector {S}
C  from {SIGAL(7-12)}
C-----
C
      CALL ZEROR(S,6)
      DO 5020 I = 1,6
        S(I) = SIGAL(I+6) + GDT*DSIGAL(I+6)
5020  CONTINUE
C
      RETURN
C
C=====
C  Format statements
C=====
C
      91  FORMAT(5X,'E11s=',E11.3)

```



```

92  FORMAT(5X,'N12S=',E11.3)
93  FORMAT(5X,'E22S=',E11.3)
94  FORMAT(5X,'N23S=',E11.3)
95  FORMAT(5X,'E33S=',E11.3)
96  FORMAT(5X,'G23S=',E11.3)
97  FORMAT(5X,'G13S=',E11.3)
98  FORMAT(5X,'G12S=',E11.3)
99  FORMAT(5X,'ALPHA11S=',E11.3)
C
143 FORMAT(10X,'Material No.=',I2,7X,'Volume Ratio = ',E10.3,/)
C
683 FORMAT(10X,'Subcell Data: '//)
684 FORMAT(1X,'(BETA | GAMMA)',2X,'SC #',2X,'SC MAT. #',2X,
&'SC MAT. TYPE',2X,'SUBCELL VOLUME'//)
685 FORMAT(2X,I2,1X,'|',I2,5X,I3,5X,I2,5X,I2,12X,E10.3)
C
866 FORMAT(///5X,'CG EFFECTIVE STIFFNESS MATRIX'//)
867 FORMAT(5X,E10.3,3X,E10.3,3X,E10.3)
868 FORMAT(44X,E10.3/
&      57X,E10.3/
&      70X,E10.3)
869 FORMAT(//5X,'EFFECTIVE ENGINEERING MODULI'//)
870 FORMAT(5X,'EFFECTIVE THERMAL EXPANSION COEFFICIENTS'//)
C
990 FORMAT(5X,'ALPHA22S=',E11.3)
991 FORMAT(5X,'ALPHA33S=',E11.3)
C
6333 FORMAT(5X,'EG',1X,6E10.3,/)
6334 FORMAT(5X,'SG',1X,6E10.3,/)
6335 FORMAT(5X,'EPG',1X,6E10.3,/)
6336 FORMAT(5X,'ETG',1X,6E10.3,/)
C
7134 FORMAT(/,5X,'The Effective Coefficients Of Thermal Expansion',/,
& 5X,'Obtained Directly From The Model (And Not Via Levin Formula)',
& /,10X,3E11.3,/)
7126 FORMAT(/,10X,'IB=',I3,2X,'IG=',I3)
7128 FORMAT(5X,6E11.4)
7129 FORMAT(19X,'Hill Concentration Tensors Of The Subcells')
7138 FORMAT(5X,'IB=',I2,2X,'IG=',I2,2X,3E11.4)

```



```

7139  FORMAT(/,7X,'Hill Subcell Thermal Concentration Vectors',/)
C
8266  FORMAT(///5X,'[CS] Effective Stiffness Matrix',/)
8267  FORMAT(5X,E10.3,3X,E10.3,3X,E10.3)
8268  FORMAT(44X,E10.3/
      &      57X,E10.3/
      &      70X,E10.3)
      END
C
C#####
SUBROUTINE BODNER(DSA,SA,STRC,NM,NS,INC,N,NN)
C
C  PURPOSE:  Bodner-Partom Viscoplastic Model
C
C  Note:  1) In This Subroutine, [SA] And [DSA] Contain The
C          "Micro" Quantities For Aboudi's Micromechanics Model
C
C          2) Arrangement Of [DSA] & [SA] Arrays
C
C          Variable          Location
C          -----
C          |
C          | Strain Rate      (1-6)
C          |-----
C          | Stress Rate      (7-12)
C          |-----
C          | Inelastic
C          | Strain Rate      (13-18)
C          |-----
C          | 12 "Slots"      (19-30)
C          | For State Variables
C          |-----
C          | Thermal Strain Rate (31-36)
C          |-----
C
C          NM - MATERIAL NUMBER
C          NS - SUBCELL NUMBER
C
C  CALLED FROM:  HYPELA (GMC)

```



```

C-----
C
      IMPLICIT REAL*8 (A-H,O-Z)
C
      LOGICAL FORDIS,HEAT,THME,NINDPT,PATRAN
C
      COMMON /MLGIC/ FORDIS,HEAT,THME,NINDPT,PATRAN
      COMMON /MICMAT/ EA(2),ET(2),GA(2),FNA(2),FNT(2),ALPA(2),ALPT(2),
&                VPBP(6,2),VPROB(2,12)
      COMMON /MICRO/ VF,MATNUM(2,2),NST,NMT,NB,NG,IDP,NSFD,NCMD
      COMMON /MICRO2/ VF1,VC1,VF2,VC2
      COMMON /MICRO3/ ICOUNT,NPLVL
      COMMON /MICRO5/ CTEMP
C
      DIMENSION SS(6),S(6),R(6),DSA(36),SA(36),STRC(6)
C
C-----
C   Extract appropriate viscoplastic material constants
C-----
C
      DO = VPROB(NM,1)
      ZO = VPROB(NM,2)
      Z1 = VPROB(NM,3)
      BM = VPROB(NM,4)
      AN = VPROB(NM,5)
      Q  = VPROB(NM,6)
C
C-----
C   Compute the deviatoric stress {S} in the subcell
C-----
C
      TEMP = (SS(1) + SS(2) + SS(3))/3.
      S(1) = SS(1) - TEMP
      S(2) = SS(2) - TEMP
      S(3) = SS(3) - TEMP
      S(4) = SS(4)
      S(5) = SS(5)
      S(6) = SS(6)
C

```



```

C-----
C Predict the average plastic strain-rate
C-----
C
AJ2=0.5*(S(1)**2+S(2)**2+S(3)**2)+S(4)**2+S(5)**2+S(6)**2
SQ3AJ = SQRT( SS(1)**2 + SS(2)**2 + SS(3)**2 +
&          2*(SS(4)**2+SS(5)**2+SS(6)**2) )
SQ2=1.414215
IF(SQ3AJ.EQ.0.) THEN
CALL ZEROR(R,6)
ELSE
R(1) = SS(1)/SQ3AJ
R(2) = SS(2)/SQ3AJ
R(3) = SS(3)/SQ3AJ
R(4) = SQ2*SS(4)/SQ3AJ
R(5) = SQ2*SS(5)/SQ3AJ
R(6) = SQ2*SS(6)/SQ3AJ
ENDIF
C
C-----
C If D0=0 then assume elastic and zero-out
C [DSA(13-30)] (inelastic strain rate and
C internal variable rates), then return
C-----
C
IF(D0.EQ.0) THEN
DO 100 JJ=13,30
DSA(JJ) = 0.0
100 CONTINUE
RETURN
C-----
C Inelastic
C-----
ELSE
C
ZEF = Z0 + Q*SA(20) +
&      (1-Q)*(R(1)*SA(21)+R(2)*SA(22)+R(3)*SA(23)+
&      R(4)*SA(24)+R(5)*SA(25)+R(6)*SA(26))
C

```



```

      IF(AJ2 .EQ. 0.) THEN
        AL=0.0
      ELSE
        ARG1=ZEF**2/(3.*AJ2)
        CON=.5*(AN+1.)/AN
        ARG=CON*(ARG1)**AN
        AL=DO/(EXP(ARG)*SQRT(AJ2))
      ENDIF
C
C-----
C  Inelastic strain rates
C-----
C
      DSA(13) = AL*S(1)
      DSA(14) = AL*S(2)
      DSA(15) = AL*S(3)
      DSA(16) = 2*AL*S(4)
      DSA(17) = 2*AL*S(5)
      DSA(18) = 2*AL*S(6)
C
C-----
C  Plastic work rate
C-----
C
      WPD = S(1)*DSA(13) + S(2)*DSA(14) + S(3)*DSA(15) +
&          S(4)*DSA(16) + S(5)*DSA(17) + S(6)*DSA(18)
C
C-----
C  State variable rates
C-----
C
      DSA(19) = WPD
      ZOM=BM/ZO
      ZD=ZOM*(Z1-ZEF)*WPD
      DSA(20)=ZD
C
      DSA(21)=ZD*R(1)
      DSA(22)=ZD*R(2)
      DSA(23)=ZD*R(3)

```



```

        DSA(24)=ZD*R(4)
        DSA(25)=ZD*R(5)
        DSA(26)=ZD*R(6)
    ENDIF
C
    RETURN
    END
C#####
    SUBROUTINE PLACE (A,B,NN)
C
    IMPLICIT REAL*8 (A-H,O-Z)
    DIMENSION A(NN),B(NN)
C
    DO 10 I=1,NN
        B(I)=A(I)
10 CONTINUE
C
    RETURN
    END
C#####
    SUBROUTINE ZEROR(A,IDIM)
C
    PURPOSE:  ZERO AN ARRAY
C
    IMPLICIT REAL*8 (A-H,O-Z)
C
    DIMENSION A(IDIM)
    DO 100 I=1,IDIM
        A(I) = 0.0
100 CONTINUE
    RETURN
    END
C#####

```


REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE March 1995	3. REPORT TYPE AND DATES COVERED Final Contractor Report	
4. TITLE AND SUBTITLE On the Finite Element Implementation of the Generalized Method of Cells Micromechanics Constitutive Model			5. FUNDING NUMBERS WU-505-63-12 NCC3-368	
6. AUTHOR(S) T.E. Wilt				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) University of Akron Department of Civil Engineering Akron, Ohio			8. PERFORMING ORGANIZATION REPORT NUMBER E-9531	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Lewis Research Center Cleveland, Ohio 44135-3191			10. SPONSORING/MONITORING AGENCY REPORT NUMBER NASA CR-195451	
11. SUPPLEMENTARY NOTES Project Manager, Steven M. Arnold, Structures Division, NASA Lewis Research Center, organization code 5220, (216) 433-3334.				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Unclassified - Unlimited Subject Categories 39 and 24 This publication is available from the NASA Center for Aerospace Information, (301) 621-0390.			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) The Generalized Method of Cells (GMC), a micromechanics based constitutive model, is implemented into the finite element code MARC using the user subroutine HYPELA. Comparisons in terms of transverse deformation response, micro stress and strain distributions, and required CPU time are presented for GMC and finite element models of a fiber/matrix unit cell. GMC is shown to provide comparable predictions of the composite behavior and requires <u>significantly</u> less CPU time as compared to a finite element analysis of the unit cell. Details as to the organization of the HYPELA code are provided with the actual HYPELA code included in the appendix.				
14. SUBJECT TERMS Micromechanics; Finite element method; Viscoplasticity; Structural analysis			15. NUMBER OF PAGES 64	
			16. PRICE CODE A04	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT	